**3D 게임프로그래밍2 과제01 설명 문서**

2017182016 박해성

1. 과제에 대한 목표
   1. 지형에서의 기본 및 디테일 텍스처 매핑 사용해보기
   2. 오브젝트 맵을 이용한 빌보드 객체 배치하기
   3. 스프라이트 애니메이션을 이용한 이펙트가 발생하는 총알 구현하기
2. 조작법

|  |  |
| --- | --- |
| ↑↓←→ | 이동키 |
| 시야 전환 | 마우스 왼쪽 클릭 후 드래그 |
| Ctrl | 총알 발사키 |

1. 구현 내용 및 코드 설명

이 프로젝트는 8-5 샘플 코드를 기반으로 제작된 코드입니다.

* 1. **Player**
* 목표: 플레이어가 Ctrl 키를 누르면 총알을 발사하게 해보자
* 가정: 총알 객체를 플레이어 객체에서 포인터 형태로 받아서 관리하기
* 실행 결과 및 설명:텍스트, 잔디, 실외이(가) 표시된 사진

  자동 생성된 설명

플레이어는 CTerrianPlayer에 기반을 두고 추가하여 만들었습니다. 플레이어는

총알 객체들을 담을 수 있는 CBulletObject\*\* 포인터를 가지고 있습니다. 총알들은 CBulletShader를 통해 Scene에서 생성되고 Scene의 CBulletObject\*\* 형의 멤버 변수에 저장됩니다. 그 후 GameFramework에서 플레이어 생성자에 Scene에 담아두었던 것을 전달합니다.

FireBullet 함수를 만들어 플레이어가 Ctrl키를 입력 시 작동하고, 플레이어 객체 생성할 때 전달받은 멤버 변수를 이용해 총알 포인터 객체에 접근 후 총알이 활성화(bool)되었는지 확인 후 활성화가 되었다면 새 총알 객체를 만들어 연결합니다. 이제 이 만들어진 객체로 총알의 초기 위치, 이동 방향, 이동 속도 등을 설정합니다.

발사 전 bool enable\_fire = true 멤버 변수를 두어 발사가능 한지를 확인하고, float recharge = 0.0f 멤버 변수를 이용해 CTerrainPlayer::Animate() 에서

recharge += fElapsedTime\*0.5f 생성 순간부터 시간을 측정하여 #define CHARGE\_TIME 1.0f 미리 설정해둔 차지 시간만큼 다음 발사 입력까지 대기합니다.

* 1. **Bullet**

총알 객체는 주로 생성 이후 이동에 관여합니다. CBulletObject::Animate() 함수에서

회전 속도, 이동 방향 등을 이용해 회전과 매 프레임 이동량과 그에 따른 위치를 설정하고 생성 시에 설정된 위치와 현재 위치와의 거리를 계산해 일정 거리를 벗어나면 렌더링 중지 및 이펙트 발생 알림을 처리하고 있습니다. (아래 코드)

텍스트이(가) 표시된 사진

자동 생성된 설명

* 1. **BillBoard**

(왼쪽 – ObjectMap.raw , 오른쪽 – 적용화면)

 지도이(가) 표시된 사진

자동 생성된 설명

텍스트, 식물, 나무이(가) 표시된 사진

자동 생성된 설명

빌보드 객체 배치에 사용되는 ObjectMap.raw를 이미지 툴을 이용하여 빌보드 객체의 분포도를 조정하였습니다. 해당 파일을 셰이더에서 읽고, 검은색은 무시되고 그외의 1~255의 회색조 값을 픽셀 단위로 읽어들여 Switch문으로 구분하여 각 객체 종류별로 수를 세어 저장하고, 그 수만큼 Width, Height 등을 각각 설정하고 생성합니다. Shader.hlsl에서는 카메라 정보를 이용하여 Look,Up, Right 벡터를 생성하고 객체의 위치 정보와 함께 월드 변환 행렬을 새로 만들어 적용합니다. 따라서 빌보드 객체는 플레이어를 항상 마주보게 됩니다.

* 1. **HeightMapTerrain**

텍스트, 무기이(가) 표시된 사진

자동 생성된 설명

디테일 텍스처에 사용할 Grass.png를 구한 후 DirectXTex툴을 이용해 DDS 파일을 생성하고, 기존 Base\_Texture.dds와 각각 샘플링한 값을 0.5씩 곱하고 더한 후에 saturate()로 클램핑하여 텍스처를 적용합니다.

* 1. **SpriteAnimaition**
* 목표: 총알이 사라지는 시점에 이펙트가 나타나도록 해보자
* 가정: 총알 객체 스프라이트 애니메이션 객체를 관리하는 세이더를 가르키는 포인터를 두고 사라지는 시점에 알리기
* 실행 결과 및 설명: 텍스트, 잔디, 자연이(가) 표시된 사진

  자동 생성된 설명

이 객체의 생성은 Scene의 BuildObject에서 CMultiSpriteObjectsShader 라는 셰이더가 생성하고 관리합니다. 위에서 말했듯이 총알 객체가 사라지는 시점에 이 객체(이펙트)가 그려집니다.

애니메이션 과정은 텍스처 생성 시 스프라이트 이미지의 사이즈 m \* n을 전달받고 AnimateRowColumn() 함수가 m\_nRow, m\_nCol 두 개의 인덱스로 전체 스프라이트 일부 중 어디를 그릴지 바꿔주는 역할을 합니다.